Subject: Re: some U++ stuff on reddit/cpp_question
Posted by Klugier on Thu, 06 Aug 2020 18:14:24 GMT
View Forum Message <> Reply to Message

Hello Xemuth,

Quote:
Maybe it is a stupid question but in what's point of having widgets from host platform instead of emulated (which allow cross platform) could be attractive for developers ?
(knowing Upp widget can work on different operating system comfort me on the idea of one time I will be able to release my application not only on windows but also on Linux)

The reason for that is that you have better look and fell in compare to Upp. The tricky part about it is that it is still multiplatform. WxWidgets creates wrapper class for that, so in application code you call wrapper not use the native code. So, your application is potable across operating systems. FileZilla is good example of multi-platform app created with that technology. The same code base for Windows, Linux and macOS. However, I always find WxWidgets application heavy and slow. It runs slows in compare to native or Upp applications.

Also, please notice that QT uses the same approach as Upp. No native widgets (KDE is the exception ;)) all is draw and the styled depending on the platform.

Quote:
from my knowledge, beginner developer don't do C++, but everytime I introduce some of them to U++ they get convinced by the power of the tool. the only problem is their lack of skills about C++ and all the complexicity of "pointers" "mutex" "stack/heap memory" . Damn you modern alternatives to C++ !

The problem with C++ is that it is 40 years old technology. Within this time humanity advanced in the term of thinking about programming language. The good example of that is Rust programming language. It is similar to C++ in context of performance (no garbage collector), but better designed. In context of memory safety (crashes are common problems with application written in C/C++) and easy to use (header files are outdated). Also it posses package manager, which makes whole platform easy to expand. Please notice that right know Linux kernel is experimenting with that technology and will probably allow to write drivers with it.

Quote:
But what can explain some framework like Qt or WxWidgets is well more known, is QT team or else has made advertisement ?

For the larger compones (corporations) QT is better that WxWidget or Upp. The main reason for that is they have money to pay for professional license. But, it is not the case. The main reason for that is they offer stability. It is develop under QT company, so in compare to Upp which is mainly maintenance by single developer - Mirek. Please imagine what will happen with Upp when Mirek will stop working on that? In case of QT they can replace employees more easily. They also offer commercial support which is nice. If you are for example blocked by issue in the library you can force them to fix that issue. In case of Upp nobody can force Mirek to do anything ;)

The features I like about QT:
- Styling via CSS (Non technical person can modify the look of the app like UX Designer - very important for bigger companies)
- QML - UI separated from the binary. You can modify layouts without recompiling application. Still important for non technical users for modifying the look of application.
- Support for mobile devices (no critical, but nice to have)
- UI can be tested - important in big projects that want to provide high quality for the users. Moreover, the risk is greater when more developers work on project
- Other IDE integration (Visual Studio, CMake, Conan packages etc).

Sincerely,
Klugier